

HEVC의 재귀적 CU 구조에 대한 조건부 확률 기반 고속 탐색 알고리즘

한 우 진^{a)†}

Conditional Probability Based Early Termination of Recursive Coding Unit Structures in HEVC

Woo-Jin Han^{a)†}

요 약

MPEG과 ITU-T에서 최근 표준화가 진행되고 있는 HEVC는 H.264/AVC에 비해, CU(coding unit), PU(prediction unit), TU(transform unit)의 다양한 형태 분할 단위를 갖는 것을 큰 특징으로 한다. 이 중, CU와 TU는 쿼드트리 형태의 재귀적 분할 구조를 가지도록 구성되는데, 압축 효율은 향상시키지만 높은 부호화 복잡도를 갖는 단점이 있다. 본 논문에서는 이러한 재귀적 분할 구조에서의 rate-distortion cost를 조건부 확률을 이용한 통계적 분석 방법을 사용하여, 분할이 일어나는 경우와 그렇지 않은 경우로 분류하는 방법을 제안한다. 제안한 방법을 HEVC의 재귀적 CU 부호화에 적용한 결과, 부호화 복잡도를 32% 가량 감소시키면서 압축 효율 하락은 0.4-0.5%로 억제할 수 있었다. 또한, HM4.0에 구현되어 있는 고속 탐색 알고리즘과 함께 사용하는 경우, 압축 효율 하락을 0.9%로 억제하면서 부호화 복잡도를 1/2로 감소시킬 수 있었다.

Abstract

Recently, High Efficiency Video Coding (HEVC) is under development jointly by MPEG and ITU-T for the next international video coding standard. Compared to the previous standards, HEVC supports variety of splitting units, such as coding unit (CU), prediction unit (PU), and transform unit (TU). Among them, it has been known that the recursive quadtree structure of CU can improve the coding efficiency while the encoding complexity is increased significantly. In this paper, a simple conditional probability to predict the early termination condition of recursive unit structure is introduced. The proposed conditional probability is estimated based on Bayes' formula from local statistics of rate-distortion costs in encoder. Experimental results show that the proposed method can reduce the total encoding time by about 32% according to the test configuration while the coding efficiency loss is 0.4%-0.5%. In addition, the encoding time can be reduced by 50% with 0.9% coding efficiency loss when the proposed method was used jointly with HM4.0 early CU termination algorithm.

Keyword : Video Coding, HEVC, Coding Unit, Rate-distortion, Fast Encoding Algorithm

1. 서 론

a) 경원대학교 소프트웨어설계및경영학과

Dept. of Software Design and Management, Kyungwon University

† 교신저자 : 한우진 (hurumi@gmail.com)

※본 연구는 2011년도 경원대학교 교내연구비 지원에 의한 연구결과로 수행되었음. (과제번호: 2011-R287)

· 접수일(2011년10월21일), 수정일(2012년2월1일), 게재확정일(2012년2월14일)

최근 MPEG과 ITU-T에서는 H.264/AVC의 압축 효율을 2배 이상 향상시키는 것을 목표로 하는 신규 영상 압축 기술인 High Efficiency Video Coding(HEVC)에 대한 표준

화를 공동으로 진행하고 있다^{[11][12]}. HEVC는 H.264/AVC와 마찬가지로 motion compensation과 DCT를 함께 사용하는 hybrid coding 방법에 기반하고 있으나, 전체 압축 시스템을 구성하는 각각의 요소 기술에서 많은 개선과 발전이 이루어져, 현재 H.264/AVC 대비 압축 효율을 약 40% 이상 향상시킬 수 있는 것으로 알려져 있다^{[3][4][5]}.

H.264/AVC에서는 하나의 slice를 16x16 크기를 갖는 매크로블록(macroblock; MB)들로 나누고, 분할모드에 따라 16x16, 16x8, 8x16, 8x8 등의 크기로 분할한 뒤, 8x8 분할의 경우 재차 8x8, 8x4, 4x8, 4x4 단위로 나눌 수 있도록 허용하는 구조를 가지고 있다. 이에 비해, HEVC에서는 최소 8x8에서 최대 64x64까지의 크기를 가질 수 있는 최대코딩단위(largest coding unit; LCU)들로 slice를 나누고, 각각의 LCU들이 쿼드트리 형태로 재귀적 분할되는 자유로운 형태를 갖고 있다. 이렇게 분할된 단위를 코딩단위(coding unit; CU)라고 하며, 이 CU들은 추가적으로 예측단위(prediction unit; PU)로 분할된다^[6].

인트라 예측 혹은 인터 예측 과정이 수행되고 나면, 원본 신호와 예측신호 간 차이인 잔차신호를 변환하는 과정을 거치게 된다. 이 과정에서, H.264/AVC의 경우 분할모드에 따라 전체 MB에 동일한 형태의 변환 크기가 사용되는 반면, HEVC에서는 CU를 기준으로 재차 재귀적 분할을 수행하고, 이 변환단위(transform unit; TU) 크기의 변환을 잔차신호에 적용한다^[7]. 이러한 CU 및 TU의 재귀적 유닛 구조는 H.264/AVC에 비해 압축 성능을 향상시키는데 있어 가장 큰 역할을 하지만, 부호화기의 복잡도가 크게 증가하는 문제점을 갖고 있다.

H.264/AVC의 경우 부호화기의 복잡도를 낮추기 위한 많은 연구들이 있어 왔다. [8]에서는 매크로블록의 rate-distortion cost를 분석하여 skip 모드로 미리 결정하기 위한 조건을 제안하였고, 인접 영역과의 예리 합을 이용하여 skip 모드 결정 성능을 향상시키는 방법도 제시되었다^[9]. 또한, [10]에서는 잔차 신호의 변환 계수가 모두 0이 되는 조건을 분석함으로써 부호화기의 복잡도를 낮출 수 있음을 보였다. 한편, 이러한 조건들은 HEVC에 적용한 경우에도, 재귀적 분할 유닛 구조 전체를 탐색하는 대신 일부분만을 탐색함

으로서 부호화기 연산량을 크게 감소시키면서 압축 성능 하락을 최소화 할 수 있다^{[11][12]}. 이러한 방법들은 매우 효과적으로 부호화기의 연산량을 감소시킬 수 있지만, 변환 계수가 모두 0인 경우에 대한 특성만을 사용하기 때문에 그렇지 않은 경우에 대한 분석 및 고속 탐색 알고리즘은 부호화기의 연산량을 추가적으로 감소시킬 수 있을 것으로 예측된다.

본 논문에서는 쿼드트리 형태의 재귀적 유닛 구조에 대해, 잔차신호의 변환 계수가 모두 0이 되는 조건 대신, rate-distortion cost 자체를 직접 조건부 확률에 기반한 통계적 분석방법을 이용하여 재귀적 분할이 일어나는 경우와 그렇지 않은 경우의 두 가지로 분류하는 방법을 제안하고, 이 방법을 HEVC의 재귀적 CU 구조에 적용하였다. 또한, 독립적으로 사용되었을 경우는 물론 HM4.0에 적용되어 있는 고속 탐색 알고리즘과 함께 적용하였을 경우의 추가 성능에 대해서도 그 결과를 보인다.

본 논문의 제 II장에서는 H.264/AVC와 HEVC의 재귀적 분할 구조에 대해 개략적으로 소개한다. 제 III장에서는 재귀적 분할 구조에 대해 적용할 수 있는 조건부 확률 기반 척도의 유도 과정을 설명하고, 제 IV장에서는 제 III장의 결과를 이용한 CU 고속 탐색 알고리즘을 설명한다. 제 V장에서 이에 대한 실험 결과를 보인 후, 마지막으로 제 VI장에서 결론을 맺는다.

II. 재귀적 코딩단위 분할

1. H.264/AVC의 비재귀적 분할구조

H.264/AVC에서는 slice를 16x16 크기를 갖는 MB들로 나눈다. 이러한 MB들은 분할모드에 따라 16x16, 16x8, 8x16, 8x8 크기 중 하나를 갖도록 분할되며, 8x8 크기를 갖는 경우 다시 8x8, 8x4, 4x8, 4x4 크기로 분할된다. 일단 분할되고 나면, profile에 따라 미리 정해진 크기의 변환이 적용된다. H.264/AVC의 baseline profile의 경우 모든 경우에 대해 4x4 크기의 블록들로 분할한 후, 4x4 크기 변환을 적용하며, H.264/AVC의 high profile에서는 인터 예측이

적용되는 16x16, 8x16, 16x8, 8x8 모드, 인트라 예측이 적용되는 8x8 모드의 경우 8x8 크기의 블록들로 분할한 후, 8x8 크기 변환을 선택적으로 적용할 수 있다. 8x8 크기 변환과 4x4 크기 변환 간 선택은 MB 단위로 이루어지며, 분할된 각각의 영역에 대해 8x8 크기 변환과 4x4 크기 변환을 선택하는 것은 허용되지 않는다.

2. HEVC의 재귀적 분할구조

HEVC에서는 slice를 최소 8x8에서 최대 64x64까지의 크기를 가질 수 있는 LCU들로 분할한다. 이 LCU는 최소 크기가 8x8이 될 때 까지 쿼드트리 형태로 재귀적 분할될 수 있으며, 이 때, 분할된 각각의 영역을 CU라고 한다. 이러한 각각의 CU들에 대해, 인트라 예측 혹은 인트라 예측 방법이 적용되며, 잔차신호가 생성된다.

한편, 이렇게 생성된 잔차신호에 대해 다시 쿼드트리 형태로 재귀적 분할이 수행되며, 이 때 분할된 각각의 영역인 TU 크기에 맞는 변환이 적용되게 된다. 이러한 유연한 분할 방법에 의해 HEVC는 H.264/AVC와는 달리 하나의 CU 내부에서 다양한 크기를 갖는 변환이 영역별로 적용되는 것을 허용한다. HEVC는 최소 4x4에서 최대 32x32까지의 크기를 갖는 변환들이 존재하므로, 하나의 CU 내에서 이러

한 변환들이 최적의 조합으로 혼합되어 사용되는 것이 가능하다. <그림 1>은 HEVC의 LCU, CU, 그리고 TU 간 관계를 보인 것이다.

III. 조건부 확률에 기반한 분류 척도

쿼드트리 형태로 표현되는 재귀적 분할 구조의 i 번째 depth에서 더 이상의 분할이 이루어지지 않는 경우를 이벤트 N 이라고 하고, 이때의 rate-distortion cost를 C 라고 하자. 이 때, 조건부 확률 $P_i(N|C)$ 는 rate-distortion cost C 가 주어졌을 때, 더 이상의 분할이 이루어지지 않는 경우의 확률이며, 반대로 조건부 확률 $P_i(N|C)$ 는 rate-distortion cost C 가 주어졌을 때, 추가적인 분할이 일어나는 경우의 확률을 의미한다. 따라서, $P_i(N|C)$ 와 $P_i(N|C)$ 를 구할 수 있다면, 두 값을 비교함으로써 간단하게 추가적인 재귀적 분할이 일어나야 하는지를 판단할 수 있다.

$P_i(N|C)$ 및 $P_i(N|C)$ 를 계산하기 위해서는 어떤 rate-distortion cost C 가 주어졌을 때, 실제 분할이 일어났는지의 여부를 통계적으로 분석해야 한다. 그러나, rate-distortion cost C 의 경우 그 값의 변이가 크고, 실수 값을 가지기 때문에 C 가 주어졌을 때의 조건부 확률을 계산하기 위한 충분

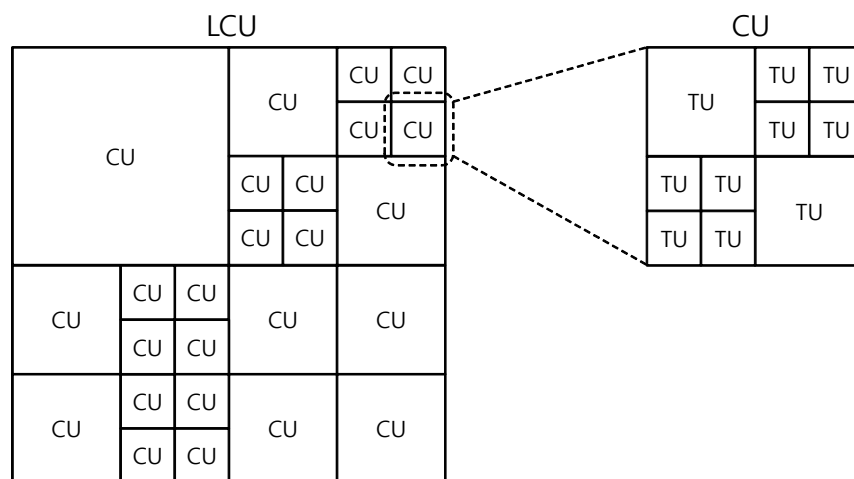


그림 1. HEVC의 LCU, CU, TU간 관계
Fig. 1. Relationship between LCU, CU and TU in HEVC

한 통계적 데이터를 확보하기가 쉽지 않다. 이에 비해, $P_i(C|N)$ 및 $P_i(C|NC)$ 의 경우, 분할이 일어났는지의 여부(N)가 주어졌을 때의 rate-distortion cost C 값을 의미하는 확률이므로 재귀적 분할 구조에서 실제 분할이 일어난 경우와 그렇지 않은 경우에 대해서 각각 통계 데이터를 모으기만 하면 쉽게 계산이 가능하다.

$P_i(N|C)$ 와 $P_i(NC|C)$ 를 $P_i(C|N)$ 과 $P_i(C|NC)$ 로부터 계산하기 위해, Bayes' formula를 적용하여 식 (1)을 얻는다.

$$P_i(MC) = \frac{P_i(C|M)P_i(M)}{P_i(C|M)P_i(M) + P_i(C|N^c)P_i(N^c)} \quad \text{and} \quad (1)$$

$$P_i(N^c|C) = 1 - P_i(MC)$$

식 (1)을 계산하기 위해서는 $P_i(C|N)$, $P_i(N)$, $P_i(C|NC)$, $P_i(NC)$ 의 네 가지 확률을 계산해야 한다. $P_i(N)$ 과 $P_i(NC)$ 는 부호화기에서 분할이 일어난 경우와 그렇지 않은 경우의 수를 셈으로써 간단하게 얻을 수 있으며, $P_i(C|N)$ 와

$P_i(C|NC)$ 의 경우 또한 분할이 일어난 경우와 그렇지 않은 경우 각각에 대한 rate-distortion cost C의 히스토그램을 계산함으로써 그 값들을 추정할 수 있다.

본 논문에서는 부호화기에서 분할이 일어난 경우와 그렇지 않은 경우 각각에 대해 수집된 rate-distortion cost C에 대해, 20개의 밴드로 구성된 히스토그램을 생성하고, 이로부터 확률을 계산하는 방법을 사용하였다. <그림 2>는 BasketballDrive 1080p@60Hz, QP 34, depth 1(32x32)에 대해, CU 재귀 트리의 부호화 과정에서 얻은 조건부 확률 $P_i(C|N)$ 과 $P_i(C|NC)$ 를 도식화 한 것이다.

<그림 2>에서 볼 수 있듯이, $P_i(C|N)$ 과 $P_i(C|NC)$ 는 rate-distortion cost C에 대해 확연히 다른 통계적 특성을 나타냄을 알 수 있다. 일단 $P_i(C|N)$, $P_i(C|NC)$, $P_i(N)$, $P_i(NC)$ 의 네 가지 확률이 계산되면, $P_i(N|C)$ 를 식 (1)로부터 계산할 수 있다. <그림 3>은 <그림 2>와 마찬가지로 경우에 대해, 식 (1)을 적용하여 계산한 $P_i(N|C)$ 값을 나타낸다.

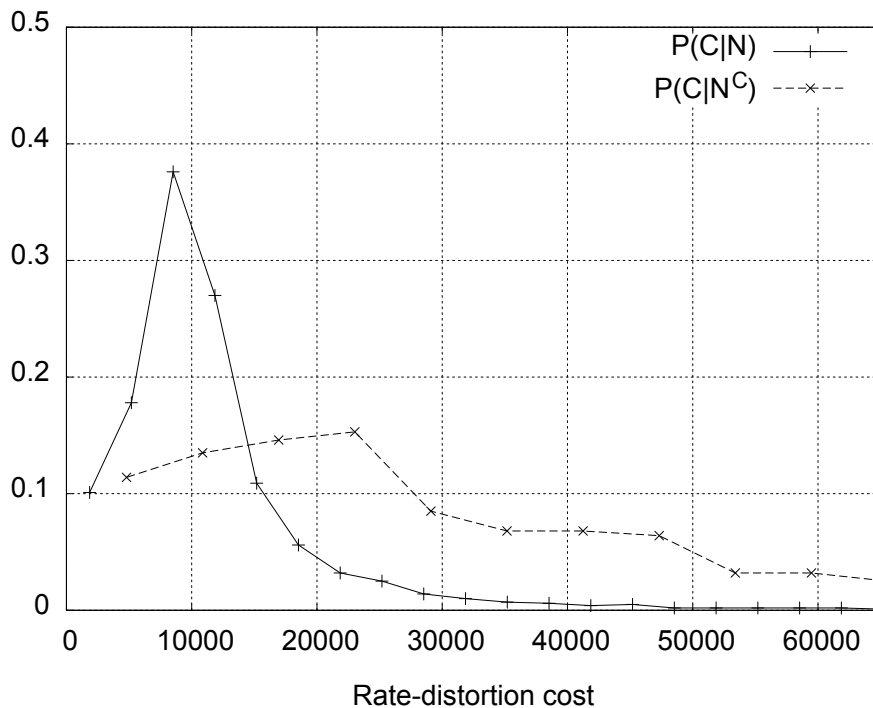


그림 2. BasketballDrive 1080p@60Hz, QP34, depth 1(32x32)에서의 조건부 확률 $P(C|N)$ 및 $P(C|NC)$
 Fig. 2. Conditional probabilities $P(C|N)$ and $P(C|NC)$ for BasketballDrive 1080p@60Hz at QP 34 and CU 32x32

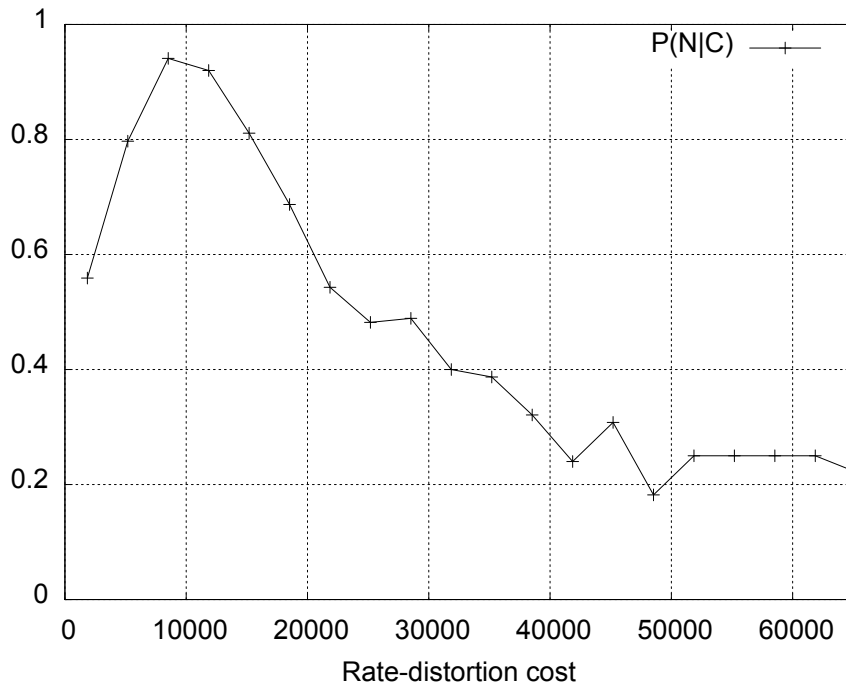


그림 3. BasketballDrive 1080p@60Hz, QP34, depth 1(32x32)에서의 조건부 확률 P(N|C)
 Fig. 3. Conditional probability P(N|C) for BasketballDrive 1080p@60Hz at QP34 and CU 32x32

$P_i(N|C)$ 와 $P_i(NC|C)$ 는 각각 rate-distortion cost C 가 주어졌을 때, 재귀적 분할이 더 이상 일어나지 않을 확률과, 재귀적 분할이 일어날 확률을 의미한다. 따라서 두 값의 크기를 비교함으로써 재귀적 분할이 추가적으로 일어나야 하는지를 판단할 수 있다. 본 논문에서는 $P_i(N|C)$ 를 $P_i(NC|C)$ 와 비교함에 있어, 압축 효율과 부호화 연산량 감소 간 trade-off 특성을 부여하기 위한 가중치 w 를 식 (2)와 같이 부여하고, $P_i(NC|C) = 1 - P_i(N|C)$ 가 성립함을 이용하여 식 (2)를 얻었다.

$$P_i(N|C) > \frac{w}{1+w} = \alpha \text{ where } w > 0 \quad (2)$$

w 의 값이 양수이기 때문에, 임계치 α 의 범위는 0과 1사이로 한정된다. 식 (2)가 성립하는 경우는 재귀적 분할이 더 이상 일어나지 않는 것을 의미하므로, 임계치 α 의 값이 커지면 재귀적 분할이 중단되는 경우가 줄어들게 되고, 즉 부호화기 연산량 감소 효과가 줄어드는 대신 압축 효율 하

락이 감소한다. 반대로, 임계치 α 의 값이 작아지면 재귀적 분할이 더욱 많이 중단되므로, 부호화기 연산량 감소 효과가 증가하면서 압축 효율 하락 폭이 커지게 된다. 이러한 방식으로, 임계치 α 를 조절함으로써 부호화기의 연산량 감소와 압축 효율 하락 간 특성을 최적화 할 수 있다. 제 V장에서, 다양한 임계치 α 의 값에 따른 연산량 감소와 압축 효율 하락 간 실험 결과를 보인다.

IV. 고속 탐색 알고리즘

제안한 방법을 적용하기 위해서는 식 (1)을 계산하기 위한 확률들인 $P_i(N)$, $P_i(NC)$, $P_i(C|N)$, $P_i(C|NC)$ 를 계산하기 위한 통계데이터를 수집해야 한다. 본 논문에서는 임의의 영상에 대한 부호화를 수행함에 있어, 1초마다 초기 5프레임을 이용하여 통계데이터를 수집하여 분석한 뒤, 이 값을 이용하여 나머지 프레임에 대해 재귀적 분할 판단을 수행하는 방식을 택하였다. 이러한 방식을 통해 별도의 오프라

인 학습 없이 제안된 방법을 적용할 수 있다.

최대 LCU 크기가 64x64라고 할 때, 재귀적 분할은 64x64 CU, 32x32 CU, 16x16 CU의 3개 depth에서 일어난다. 이를 depth 0, depth 1, depth 2라고 하면, $P_i(N)$ 과 $P_i(NC)$ 를 계산하기 위해서는 각 i 번째 depth에 대해, 재귀적 분할이 일어나지 않은 횟수 C_N^i 와, 총 비교 횟수인 C_T^i 를 누적하여 계산한다. 이로부터, i 번째 depth에 대한 확률 $P(N)$ 과 $P(NC)$ 는 식 (4)와 같이 계산된다.

$$P_i(N) = \frac{C_N^i}{C_T^i} \text{ and } P_i(N^C) = 1 - P_i(N) \quad (4)$$

$P_i(C|N)$ 과 $P_i(C|NC)$ 를 계산하기 위해서는 재귀적 분할이 일어난 경우와 그렇지 않은 경우의 rate-distortion cost들에 대한 히스토그램을 생성해야 한다. 본 논문에서는 각 depth 별로 두 가지 경우에 대해 각각 rate-distortion cost를 수집한 후, 최소값과 최대값을 산출하고, 이 두 값 사이를 균등한 길이로 20등분한 히스토그램을 사용하였다. 즉, i 번째 depth에 대해, 재귀적 분할이 일어나지 않은 경우의 rate-distortion cost들에 대한 데이터를 수집한 후, 이 데이터가 어떤 대역에 속하는 지를 계산하고, 해당되는 대역에 대한 누적 횟수를 증가시킴으로써 히스토그램 $H_{Ni}[k]$ 를 계산한다.

일단 히스토그램 $H_{Ni}[k]$ 가 주어지면, 임의의 rate-distortion cost C 가 주어졌을 때 i 번째 depth에 대한 조건부 확률 $P_i(C|N)$ 은 식 (5)와 같이 계산된다.

$$P_i(C|N) = H_N^i[k] / \sum_{k=0}^{19} H_N^i[k]$$

where

$$k = \max\left(0, \min\left(19, \left\lfloor \frac{C^i - MinC^i}{S^i} + 0.5 \right\rfloor\right)\right) \quad (5)$$

and

$$S^i = \left\lfloor \frac{MaxC^i - MinC^i}{20} + 0.5 \right\rfloor$$

식 (5)에서 $MaxC^i$ 와 $MinC^i$ 는 각각 i 번째 depth에 대해, 재귀적 분할이 일어나지 않은 경우 수집된 rate-distortion

cost값들의 최대 및 최소값이다. $P_i(C|NC)$ 의 경우는 재귀적 분할이 발생한 경우에 대해 수집한 rate-distortion cost들의 히스토그램을 이용한다는 점만 다를 뿐 다른 과정은 동일하다. <그림 4>는 이러한 과정들을 그림으로 나타낸 것이다.

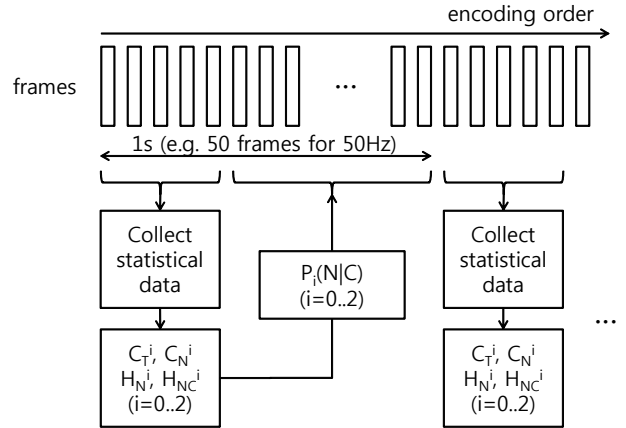


그림 4. 제안한 고속 탐색 알고리즘
Fig. 4. Proposed fast encoding algorithm

V. 실험 결과

1. 테스트 조건

제안한 방법의 효과를 검증하기 위해, HM4.0[14]에 제안한 방법을 구현하고, 압축 효율 및 부호화 연산량 측면에서 비교하였다. 특히, HM4.0에 이미 구현되어 있는 고속 탐색 알고리즘^[11]을 함께 적용한 경우에 대해서도 실험 결과를 구하였다. 압축 효율 및 부호화 연산량을 비교하기 위해서 JCT-VC에서 사용하는 공통 테스트 조건^[15] 중 random-access scenario와 low-delay scenario를 사용하였으며, 평가 영상으로는 [15]에서 사용되는 테스트 영상 중 부호화 연산량이 가장 크게 문제가 되는 5개의 1080p 영상인 Kimono, ParkScene, Cactus, BasketballDrive, BQTerrace를 사용하였다. HM4.0을 기준으로 상대적 변화를 측정하기 때문에 BD-rate^[16]에 의해서 구한 동일 PSNR 대비 비트 증가율이 양수일 경우 압축 효율이 하락하였음을 의미하며,

실행 속도가 100%보다 작을 경우 부호화기 연산량이 절감되었음을 의미한다.

2. 제안한 방법의 성능

<표 1>과 <표 2>는 식 (3)의 임계치 α 값을 0.5, 0.6, 0.7, 0.8로 변화시키면서 제안한 방법의 효과를 평가한 결과이다.

<표 1>과 <표 2>의 결과로부터, 임계치의 값이 증가됨에 따라 부호화기 연산량 감소 효과가 감소하며 대신 압축 효율 감소가 완화되는 것을 알 수 있다. 구체적으로, 임계치의 값이 0.5인 경우에는 부호화기 연산량이 42%-45%가량 감

소하며, 압축 효율 또한 3.3%-3.6% 가량 하락하였다. 한편 임계치의 값을 0.8로 설정한 경우, 부호화기 연산량이 약 32% 감소하면서도 압축 효율 하락이 0.4%-0.5% 가량으로 억제되어 제안한 방법이 효과적으로 동작함을 알 수 있었다. 이러한 결과로부터, 이후의 실험에서는 임계치를 0.8로 설정하고 진행하였다.

3. HM4.0에 적용되어 있는 고속 알고리즘과의 비교

HM4.0에는 잔차신호의 변환계수가 모두 0인 경우에 대한 조건을 이용하는 고속 알고리즘^{[11][12]}이 구현되어 있다. 이 중, [11]의 방법에 대해, 제안한 방법과 각각 적용한 결

표 1. 다양한 임계치에 따른 제안한 방법의 성능 (random access scenario) (동일 PSNR 대비 비트율 증가 (%) / 부호화기 연산량 (%)으로 표시)
Table 1. Performances of random access scenario according to various thresholds (BD-rate increases (%) / relative running time (%))

Sequence	$\alpha=0.5$	$\alpha=0.6$	$\alpha=0.7$	$\alpha=0.8$
Kimono	1.8% / 56%	1.3% / 60%	0.8% / 63%	0.4% / 67%
ParkScene	2.3% / 61%	1.8% / 63%	0.8% / 66%	0.2% / 70%
Cactus	5.1% / 60%	4.1% / 63%	1.8% / 67%	0.6% / 70%
BasketballDrive	4.7% / 57%	2.3% / 60%	1.2% / 63%	0.4% / 66%
BQTerrace	2.5% / 56%	1.6% / 59%	1.1% / 62%	0.3% / 65%
Average	3.3% / 58%	2.2% / 61%	1.1% / 64%	0.4% / 68%

표 2. 다양한 임계치에 따른 제안한 방법의 성능 (low-delay scenario) (동일 PSNR 대비 비트율 증가 (%) / 부호화기 연산량 (%)으로 표시)
Table 2. Performances of low-delay scenario according to various thresholds (BD-rate increases (%) / relative running time (%))

Sequence	$\alpha=0.5$	$\alpha=0.6$	$\alpha=0.7$	$\alpha=0.8$
Kimono	2.4% / 47%	1.8% / 52%	1.0% / 58%	0.5% / 63%
ParkScene	3.4% / 63%	2.4% / 66%	1.1% / 69%	0.4% / 73%
Cactus	5.7% / 58%	3.3% / 63%	2.1% / 67%	0.8% / 72%
BasketballDrive	3.7% / 55%	2.7% / 58%	1.5% / 61%	0.5% / 66%
BQTerrace	2.7% / 55%	1.7% / 59%	0.9% / 66%	0.4% / 69%
Average	3.6% / 55%	2.4% / 60%	1.3% / 64%	0.5% / 68%

표 3. HM4.0 고속 탐색 알고리즘과의 비교 (random access scenario) (동일 PSNR 대비 비트율 증가 (%) / 부호화기 연산량 (%)으로 표시)
Table 3. Comparison with HM4.0 fast algorithm for random access scenario (BD-rate increases (%) / relative running time (%))

Sequence	[11]	제안한 방법 ($\alpha=0.8$)	[11]+제안한 방법
Kimono	0.2% / 68%	0.4% / 67%	0.6% / 52%
ParkScene	0.5% / 57%	0.2% / 70%	0.7% / 46%
Cactus	0.5% / 62%	0.6% / 70%	1.1% / 50%
BasketballDrive	0.7% / 64%	0.4% / 66%	1.1% / 49%
BQTerrace	0.5% / 58%	0.3% / 65%	0.8% / 44%
Average	0.5% / 62%	0.4% / 68%	0.9% / 48%

표 4. HM4.0 고속 탐색 알고리즘과의 비교 (low-delay scenario) (동일 PSNR 대비 비트율 증가 (%) / 부호화기 연산량 (%)으로 표시)
 Table 4. Comparison with HM4.0 fast algorithm for low-delay scenario (BD-rate increases (%) / relative running time (%))

Sequence	[11]	제안한 방법 ($\alpha=0.8$)	[11]+제안한 방법
Kimono	0.3% / 73%	0.5% / 63%	0.8% / 53%
ParkScene	0.4% / 60%	0.4% / 73%	0.8% / 49%
Cactus	0.7% / 65%	0.8% / 72%	1.4% / 53%
BasketballDrive	0.2% / 67%	0.5% / 66%	0.8% / 51%
BQTerrace	0.5% / 59%	0.4% / 69%	0.9% / 47%
Average	0.4% / 65%	0.5% / 68%	0.9% / 51%

과, 그리고 함께 적용한 결과를 <표 3>과 <표 4>에 보였다.

<표 3>과 <표 4>에서 볼 수 있듯이 [11]의 방법과 제안한 방법은, 각각 적용하였을 경우 부호화기 연산량을 32%-38% 가량 낮출 수 있으며 이 때 0.4-0.5% 정도의 압축 효율 하락이 관측된다. [11]의 방법과 제안한 방법을 함께 적용한 경우, 부호화기 연산량은 49%-52% 가량 낮아지며, 이 때 압축 효율 하락은 0.9%로서, 제안한 방법을 [11]의 방법과 함께 적용하면, 1% 이내로 압축 효율 하락을 억제하면서도 HEVC의 부호화기 연산량을 약 1/2로 감소시킬 수 있는 것을 확인할 수 있었다.

VI. 결 론

본 논문에서는 쿼드트리 형태의 재귀적 분할 구조에 있어, rate-distortion cost를 조건부 확률에 기반한 통계적 분석 방법에 의해 분류하는 방법을 제안하고, 이에 기반한 고속 탐색 알고리즘을 적용하였다. 제안한 방법을 HEVC의 재귀적 CU 분할 구조에 적용한 결과 부호화기 연산량을 32% 감소시키면서 압축 효율 하락을 0.4-0.5%로 억제할 수 있다. 한편, 기존 HM4.0에 적용되어 있는 고속 탐색 알고리즘과 함께 사용하는 경우, 압축 효율 하락을 0.9%로 억제하면서 부호화기 연산량을 약 1/2로 감소시킬 수 있었다.

참 고 문 헌

[1] G. J. Sullivan, J.-R. Ohm, "Recent Developments in Standardization of High Efficiency Video Coding (HEVC)", SPIE Applications of Digital Image Proc. XXXIII, Proc. SPIE, Vol. 7798, paper 7798-30, Aug.

2010.
 [2] JCT-VC, "Report of Subjective Test Results of Responses to the Joint Call for Proposals (CFP) on Video Coding Technology for High Efficiency Video Coding (HEVC)," Document JCTVC-A204, Dresden, DE, Apr. 2010.
 [3] E. Ohwovoriole, Y. Andreopoulos, "Rate-distortion performance of contemporary video codecs: Comparison of Google/WebM VP8, AVC/H.264, and HEVC TMuC", LENS Symp., London, Sep. 2010.
 [4] F. De Simone, L. Goldmann, J.-S. Lee, T. Ebrahimi, "Performance analysis of VP8 image and video compression based on subjective evaluations," SPIE Appl. Digital Image Proc. XXXIV, Aug. 2011.
 [5] T. Wiegand, J.-R. Ohm, G. J. Sullivan, W. J. Han, R. Joshi, T. K. Tan and K. Ugur, "Special Section on the Joint Call for Proposals on High Efficiency Video Coding (HEVC) Standardization," IEEE Transactions on Circuits and Systems for Video Technology, Vol. 20, No. 12, pp. 1661-1666, Dec. 2010.
 [6] Woo-Jin Han, Junghye Min, Il-Koo Kim, Elena Alshina, Alexander Alshin, Tammy Lee, Jianle Chen, Vadim Seregin, Sunil Lee, Yoon-Mi Hong, Min-Su Cheon, Nikolay Shlyakhov, Ken McCann, Thomas Davies and Jeong-Hoon Park, "Improved video compression efficiency through flexible unit representation and corresponding extension of coding tools," IEEE Transactions on Circuits and Systems for Video Technology, Vol. 20, No. 12, pp. 1709-1720, Dec. 2010.
 [7] D. Marpe, H. Schwarz, S. Bosse, B. Bross, P. Helle, T. Hinz, H. Kirchhoffer, H. Lakshman, T. Nguyen, S. Oudin, M. Siekmann, K. Suhring, M. Winken, and T. Wiegand, "Video Compression Using Nested Quadtree Structures, Leaf Merging, and Improved Techniques for Motion Representation and Entropy Coding," IEEE Transactions on Circuits and Systems for Video Technology, Vol. 20, No. 12, pp. 1676-1687, Dec. 2010.
 [8] C. S. Kannangara, E. G. Richardson, M. Bystrom, J. R. Solera, Y. Zhao, A. Maclennan, R. Cooney, "Low complexity skip prediction for H.264 through Lagrangian cost estimation," IEEE Transactions on Circuits and Systems for Video Technology, Vol. 16, No. 2, pp. 202-208, Feb. 2006.
 [9] I. Choi, J. Lee, B. Jeon, "Fast coding mode selection with rate-distortion optimization for MPEG-4 Part 10 AVC/H.264," IEEE Transactions on Circuits and Systems for Video Technology, Vol. 16, No. 12, pp. 1557-1561, Dec. 2006.
 [10] H. Wang, S. Kwong, C. W. Kok, "An efficient mode decision algo-

- rithm for H.264/AVC encoding optimization," IEEE Transactions on Multimedia, Vol. 9, No. 4, pp. 882-888, Jun. 2007.
- [11] R. H. Gweon, Y. L. Lee, J. Lim, "Early termination of CU encoding to reduce HEVC complexity," JCTVC-F045, 6th JCT-VC meeting, Jul. 2011, Torino, Italy.
- [12] K. Choi, S. H. Park, E. S. Jang, "Coding tree pruning based CU early termination," JCTVC-F092, 6th JCT-VC meeting, Jul. 2011, Torino, Italy.
- [13] B. Bross, W. J. Han, J.-R. Ohm, G. J. Sullivan, T. Wiegand, "WD4: Working Draft 4 of High-Efficiency Video Coding," Document JCTVC-F803, Torino, IT, July 2011.
- [14] HEVC Model 4.0, available for download at <http://hevc.kw.bbc.co.uk/trac/browser/jctvc-hm/tags/HM-4.0>.
- [15] F. Bossen, "Common test conditions and software reference configurations," Document JCTVC-F900, Torino, IT, July 2011.
- [16] G. Bjontegarrd, "Calculation of average PSNR differences between RD curves," in ITU-T SC16/Q6 13th VCEG meeting, No. VCEG-M33, Austin, TX, Apr. 2001.

저 자 소 개



한 우 진

- 1995년 2월 : KAIST 전산학과 공학사
- 1997년 2월 : KAIST 전산학과 공학석사
- 2002년 2월 : KAIST 전산학과 공학박사
- 2002년 3월 ~ 2003년 3월 : SL2 연구소장
- 2003년 4월 ~ 2011년 8월 : 삼성전자 DMC 연구소 수석연구원
- 2011년 9월 ~ 현재 : 강원대학교 소프트웨어설계경영학과 교수
- 주관심분야 : 영상압축, 영상이해, 멀티미디어통신